類似ドキュメントの発見手法の検討

4K－9

青木 圭子　松本 一則　橋本 和夫
国際電信電話株式会社 研究所

1. はじめに

インターネットはドキュメントを自由に追加・更新できるオープンな環境であるため，欲しい情報を一定期間に渡って探し続ける機能が必要である．しかし，従来型の検索エンジンでは事前に収集しておいたドキュメントを対象に検索を行うため，ある時刻での情報しか得ることができない．そこで，収集済みのドキュメントだけでなく，情報発信者が新たに追加・更新したドキュメントの中から指定されたドキュメントに類似したものを発見し，随時通知する検索エンジンを検討している．

本稿では，同エンジンを実現する上で問題となる，大量ドキュメントの分類・更新手法について述べる．

2. 従来のドキュメント分類手法と実装上の問題点

本稿で検討対象とする検索エンジンでは，バッチ処理による検索が前提となるため，キーワードによる検索ではなく，検索対象となるドキュメントを予めクラスタリングし，ユーザが指定したドキュメントに近い内容のドキュメントを検索する手法を用いることとする．ドキュメントのクラスタリングに関しては，語の出現頻度を特徴量とし，クラスタの全メンバの事後確率を最大化する手法があり，一般的なクラスタリング手法であるWard's法と比較して，より正確な分類を行うことが知られている[1]．

本章では同手法の簡単な説明とそれを大量ドキュメントに適用した場合の問題点について述べる．

2.1 事後確率を用いたクラスタリング

事後確率を用いたドキュメントのクラスタリングは以下のような流れになる．

(1) 各ドキュメントを1クラスタとする．

(2) クラスタ c_i と c_j をマージしたクラスタ c を仮定したときの，c の語分布のもとで c_i と c_j の語分布が起こる事後確率 $P(c \mid c_i, c_j)$ を全クラスタの組み合わせについて求め，事後確率が最大となる組み合わせをマージし，1クラスタにする．

(3) クラスタ数が1になるまで(2)を繰り返す．

A Consideration on Discovering Similar Documents
Keiko AOKI, Kazunori MATSUMOTO, Kazuo HASHIMOTO
KDD R&D Laboratories
2-1-15 Ohara, Kamifukuoka, Saitama 356, Japan

2.2 大量ドキュメントに適用する場合の問題点

前記クラスタリングのアルゴリズムを用いて大量のドキュメントを分類する場合，以下の問題が生じる．

(1) クラスタを生成する際，全クラスタの総当たりで事後確率を求めるため，ドキュメント数を n とすると事後確率の総計算量が $O(n^3)$ となる．このため，膨大な数のドキュメントを扱わなければならない検索エンジンの場合，ドキュメントの数が増えるに従ってクラスタ生成が難しくなる．

(2) 新たにドキュメントが収集された場合，クラスタを更新する必要がある．従来アルゴリズムではクラスタの一括生成を対象としており，更新の際に作成済のクラスタを利用できないため，クラスタの更新処理に時間がかかる．

上記問題を解決するため，以下の提案を行う．

3. 大量ドキュメントのための分類・更新手法

3.1 分類のためのクラスタリング手法

3.1.1 概要

大量のドキュメントをより少ない計算量で分類するため，以下のようなクラスタリングの手法を提案する．

- ドキュメント数がシステムの定めた最大値(MAX)未満の場合，従来のアルゴリズムを用いてクラスタリングを行う．
- ドキュメントの数が MAX を越えた場合，

  (1) 後述 (**3.1.2**) の評価関数を用いて，最適な MAX 個のドキュメントの集合を求める．

  (2) 得られたドキュメント集合からクラスタを作成し，残りのドキュメントをそのクラスタの葉ノードを基準にして分類する．図1に $MAX = 4$ の場合の例を示す．ここでは，1,2,3,4のドキュメントが最初に選択され，各葉ノードに残りのドキュメントを割り当てる．

  (3) 各葉ノードについて，葉ドキュメントと割り当てられたドキュメントに対して再帰的にクラスタリングを行い，そのクラスタのルートノードを元の葉ノードと置き換える．図2の例では，図1の葉ノード4に割り当てられたドキュメントと葉ドキュメントを再帰的にクラスタリングし，置き換える．

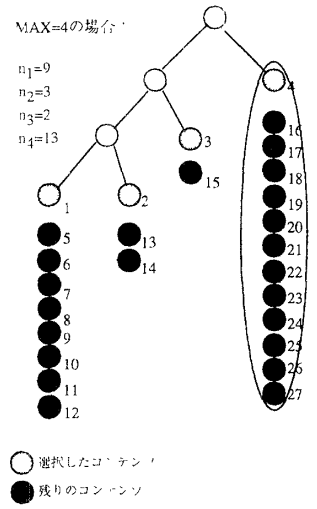

図 1: ドキュメント割り当ての例

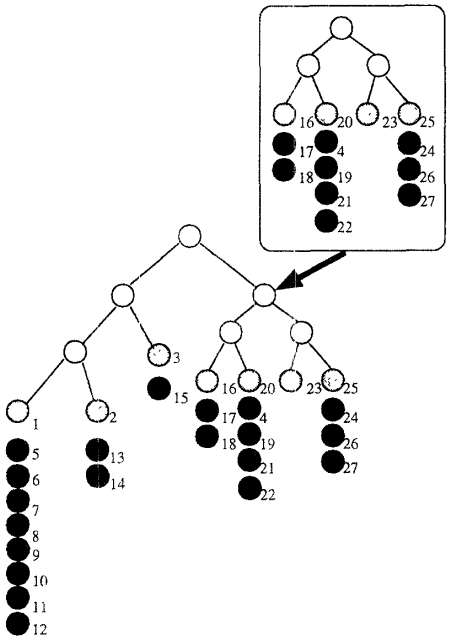

図 2: 再帰的クラスタリングの例

3.1.2 最適化のための評価関数

ドキュメント集合の最適化では MDL 基準に基づき，分類結果の符号長が最小になるようにする．

分類結果の符号長 (L) は，木の記述自体に必要な情報量 (L_1) と与えられたデータに対する対数尤度 (L_2) の和として計算する．

木自体の符号化は木を先行順 (preorder) に訪れて内部ノードを訪れたときに 1 を出力し，端点を訪れたときに 0 を出力することによって行う[2]．この場合，葉ノードの数 ($= MAX$) を k とすると，内部ノードの数は $k-1$ となり，$L_1 = 2k-1$ となる．

残りのドキュメントは，最も近い葉ノードに割り当てる．葉ノード i に割り当てられたドキュメントの数を n_i，葉ノード i がそのノードに選択される確率を p_i とすると，与えられたデータに関する記述長 L_2 は式 (1) のようになる．

$$L_2 = -\sum_i^k n_i log p_i = -\sum_i^k n_i log \frac{n_i}{\sum_j n_j} \qquad (1)$$

3.2 クラスタの更新手法

ここでは，ドキュメントが更新された場合，クラスタを再構築するのではなく，もとのクラスタの形を残したまま，最も近いノードにそのドキュメントに対応するノードを追加する．具体的には，

(1) 新しいドキュメント x を収集したとき，ドキュメント x の頻度表を作成する．既存クラスタのルートノードを y とする．

(2) x を y の子ノード (左) または y の子ノード (右) とマージしたときの事後確率を，各子ノードの頻度表 (クラスタ生成時に計算しておく) を用いて計算する．事後確率が大きい方のノードを z とする．

(3) z が葉ノードの場合，z と x を葉ノードとする部分木をかつての z の位置にあった葉ノードと置き換える．z が中間ノードの場合，z を y として (2) から繰り返す．

本更新手法を用いた場合，クラスタ更新の計算量は $O(log\ n)$ となる．この手法では，ノードが本来あるべき位置と異なる場合があるため，定期的にクラスタの再構築を行う．

4. おわりに

本稿では大量のデータを対象とした，ドキュメントのクラスタリングとクラスタの更新手法について検討した．これにより，情報源から適時収集するドキュメントをインクリメンタルにクラスタリングし，事前に指定されたドキュメントに類似するものが新たに発生したことを検出することができる．今後は，これらの検討結果をもとにシステムを試作し，評価を行う予定である．

参考文献

[1] Makoto IWAYAMA, Takenobu TOKUNAGA, "Hierarchical Bayesian Clustering for Automatic Text Classification", Proceedings of IJCAI-95, pp.1322-1327, 1995.

[2] 伊藤, 川端, "パラメタ分散推定量を用いたユニバーサル・データ圧縮アルゴリズム", 第 8 回情報理論とその応用研究会, p.239-244, 1985.